

# Trabalho 3 - Processamento de Imagens

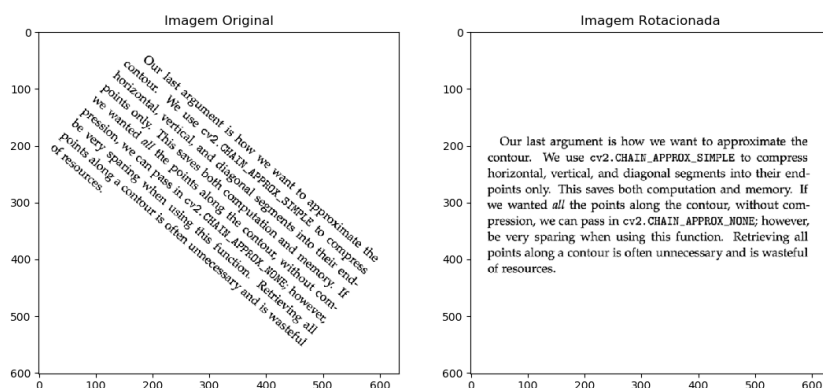
Décio Gonçalves de Aguiar Neto<sup>1</sup>

<sup>1</sup>Universidade Estadual de Campinas - UNICAMP  
Instituto de Computação

## 1. IMPLEMENTAÇÃO

O objetivo deste trabalho é aplicar os métodos descritos no enunciado para encontrar o ângulo necessário para rotacionar a imagem de forma que a mesma fique alinhada, uma outra observação a ser feita é que além de ser alinhada a imagem deve passar por um processo de interpolação para que fique preenchida e de forma inteira. A Figura 1 apresenta um antes e depois de aplicada a rotação, e após aplicada a rotação feita a interpolação sobre a imagem rotacionada.

Figura 1. Antes e Depois de aplicada a rotação e interpolação.



Para a implementação deste trabalho foi utilizada a biblioteca MISC para o carregamento das imagens em níveis de cinza a fim que pudessem ser aplicados os procedimentos necessários para o alinhamento das imagens. Os filtros de bordas utilizados foram importados da biblioteca Skimage e os métodos de alinhamento implementados seguiram o pseudocódigo fornecido no enunciado.

## 2. EXECUÇÃO

Para que o código seja executado de forma correta basta executar os seguintes comandos:

```
python3 alinhar.py caminho_da_imagem modo imagem_saida.png
```

A entrada do modo deve ser 0 ou 1. No modo 0 o algoritmo utilizado será o baseado na transformada de Hough, e no modo 1 será o algoritmo baseado em projeção horizontal.

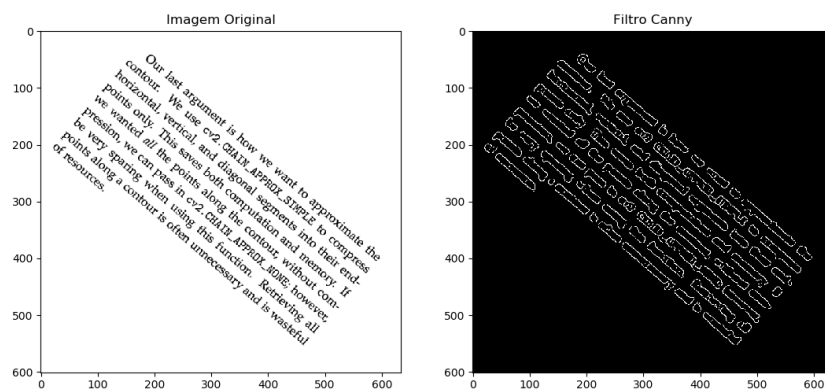
A saída será dada da seguinte forma, será exibido o ângulo encontrado para realizar o alinhamento no console e também a imagem resultante após o alinhamento.

### 3. RESULTADOS

Nesta seção serão descritos os resultados obtidos, e os passos intermediários que levaram à implementação final dos algoritmos utilizados.

Para a construção do método de alinhamento utilizando projeção horizontal, a implementação seguiu exatamente o pseudocódigo fornecido no enunciado e tendo como função objetivo a soma dos quadrados das diferenças dos valores em células adjacentes dos histogramas gerados a partir das imagens de entrada. Durante os testes realizados logo se percebeu que essa simples implementação não obteve resultados satisfatórios para todo o conjunto de testes, assim sendo necessário realizar modificações a fim de tornar o método mais robusto. Para trazer melhorias nesse método, foram utilizados alguns filtros, dentre eles Sobel, Laplaciano e Canny, que são filtros de detecção de bordas fornecidos pela biblioteca Skimage. O filtro de bordas que obteve melhores resultados sobre todo o conjunto de teste foi o Canny com sigma igual a 3. A Figura 2 apresenta a imagem original e seu mapa de bordas gerado utilizando o filtro Canny.

**Figura 2. Imagem e seu mapa de bordas.**



Para a implementação do método de alinhamento utilizando a transformada de Hough, foi utilizado o método `Hough_lines` da biblioteca Skimage, que retorna a matriz de acumuladores dos ângulos e distâncias obtidas. Tendo esses dados, foi feita uma busca pela coluna onde se encontra o valor máximo de acúmulo na matriz de acumuladores e o índice dessa coluna foi utilizado para obter o valor contido no mesmo índice do vetor de ângulos que contém o ângulo desejado, onde o maior número de linhas se interceptam. A Figura 3 apresenta a imagem original e sua matriz de acumuladores da transformada de Hough.

Figura 3. Imagem e sua Transformada de Hough.

